



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
13/226,079	09/06/2011	Richard Harold Boivie	YOR920110542US1	2909

48150 7590 05/01/2017
MCGINN INTELLECTUAL PROPERTY LAW GROUP, PLLC
8321 OLD COURTHOUSE ROAD
SUITE 200
VIENNA, VA 22182-3817

EXAMINER

AVERY, JEREMIAH L

ART UNIT	PAPER NUMBER
----------	--------------

2431

MAIL DATE	DELIVERY MODE
-----------	---------------

05/01/2017

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Ex parte RICHARD HAROLD BOIVIE and
DIMITRIOS PENDARAKIS

Appeal 2016-004345
Application 13/226,079¹
Technology Center 2400

Before JASON V. MORGAN, JOSEPH P. LENTIVECH, and
MICHAEL J. ENGLE, *Administrative Patent Judges*.

MORGAN, *Administrative Patent Judge*.

DECISION ON APPEAL

Introduction

This is an appeal under 35 U.S.C. § 134(a) from the Examiner's Final Rejection of claims 1–24. We have jurisdiction under 35 U.S.C. § 6(b).

We REVERSE.

¹ Appellants identify International Business Machines Corporation as the real party in interest. App. Br. 1.

Invention

Appellants disclose a central processing unit that is capable of executing an EnterSecureMode instruction that enables secure object information retrieved from and stored into a memory to be respectively decrypted and encrypted. Abstract; Spec. 9.

Exemplary Claim

Claim 1, reproduced below with key limitations emphasized, is illustrative:

1. An apparatus, comprising:

a memory to store a secure object comprising at least one of code and data that is encrypted when stored in said memory; and

a central processing unit (CPU) capable of executing an EnterSecureMode (esm) instruction that enables decryption of information in the secure object when the secure object information is retrieved from said memory into said CPU and encryption of the secure object information when the secure object information exits said CPU to be written back into said memory, to thereby automatically cryptographically protect said secure object information when said secure object information exits said CPU,

said CPU further comprising a feature to protect said secure object from code that the secure object receives from other software, as based on using information contained in said secure object itself.

Rejections²

The Examiner rejects claims 1–5 and 7–24 under 35 U.S.C. § 103(a) as being unpatentable over Kocher (US 8,055,910 B2; Nov. 8, 2011) and Sahita (US 8,381,288 B2; Feb. 19, 2013). Final Act. 7–13.

The Examiner rejects claim 6 under 35 U.S.C. § 103(a) as being unpatentable over Kocher, Sahita, and Hunter (US 2007/0006294 A1; Jan. 4, 2007). Final Act. 13–15.

ANALYSIS

Issue: Did the Examiner err in finding the combination of Kocher and Sahita teaches or suggests “a central processing unit (CPU) capable of executing an EnterSecureMode (esm) instruction that enables . . . encryption of the secure object information when the secure object information exits said CPU to be written back into said memory,” as recited in claim 1?

In rejecting claim 1, the Examiner finds Kocher’s use of query cryptographic oracles 230 to decrypt content or code portions teaches or suggests *a central processing unit (CPU) capable of executing an EnterSecureMode instruction that enables encryption of the secure object information when the secure object information exits said CPU to be written back into said memory*. Final Act. 8 (citing Kocher Fig. 2, col. 8, ll. 20–42, 53–62). In particular, the Examiner finds that the automatic cryptographic protection of secure object information that exits the CPU is an obvious design choice. Final Act. 8–9.

² Claims 1–24 were also rejected under 35 U.S.C. § 101. Final Act. 5–6. However, the Examiner withdrew this rejection. Ans. 2. Thus, we do not address the merits of this rejection.

Appellants contend the Examiner erred because “the ‘secure object’ information in Kocher does not get returned again into memory 200, let alone get encrypted again as it leaves the CPU for return to memory 200.” App. Br. 16. Appellants further argue that Kocher does not provide support for the Examiner’s finding that such encryption would be an obvious design choice. *Id.* at 18.

Appellants’ arguments are consistent with the cited teachings of Kocher, which depict a bulk decryption module 240 that provides data through output interface 250 to destination program or device 260, but do not similarly depict or otherwise detail the encryption of information being written back into a memory such as media 200. *See, e.g.*, Kocher Fig. 2.

In response, the Examiner merely identifies additional teachings in Kocher related to ensuring a player or code is authorized to read or modify content. Ans. 5 (citing Kocher col. 15, ll. 24–55, l. 27, ll. 36–50, col. 30, ll. 26–35, and col. 32, ll. 57–59). However, the Examiner does not show that these teachings relate to the encryption of secure object information exiting a CPU to be written back into a memory in the manner claimed.

The Examiner concludes that the claimed EnterSecureMode instruction should be interpreted broadly as encompassing “Kocher’s disclosure of function calls and other such instructions in carrying out the cryptographic and other security functions” (Ans. 5–6) and that “the claim language of ‘to *thereby* automatically cryptograph[ically] protect said secure object information . . . ’ is an intended use” (*id.* at 6). However, the Examiner’s findings and conclusions do not address the disputed recitation limiting the meaning of the claimed EnterSecureMode instruction such that it *enables encryption of the secure object information when the secure object*

information exits said CPU to be written back into said memory. The “enables encryption” recitation is not merely an intended use of the claimed EnterSecureMode instruction; rather, it defines in-part what the EnterSecureMode instruction does. The Examiner’s interpretation of the EnterSecureMode recitation, by failing to address the limiting features of the recitations that precede the *thereby* clause, is not supported by the cited evidence. Therefore, the Examiner’s findings do not show that Kocher teaches or suggests the disputed recitation.

Furthermore, the Examiner does not rely on Sahita to cure the noted deficiency of Kocher. Therefore, we agree with Appellants that the Examiner’s findings do not show the combination of Kocher and Sahita teaches or suggests “a central processing unit (CPU) capable of executing an EnterSecureMode (esm) instruction that enables . . . encryption of the secure object information when the secure object information exits said CPU to be written back into said memory,” as recited in claim 1.

Accordingly, we do not sustain the Examiner’s 35 U.S.C. § 103(a) rejection of claim 1, and claims 2–5 and 7–24, which contain similar recitations. The Examiner also does not show that Hunter cures the noted deficiency of Kocher and Sahita. Therefore, we also do not sustain the Examiner’s 35 U.S.C. § 103(a) rejection of claim 6.

DECISION

We reverse the Examiner’s decision rejecting claims 1–24.

REVERSED